arXiv:1110.2324v1 [math.NA] 11 Oct 2011

Relative Error Control in Bivariate Interpolatory Cubature

J.S.C. Prentice
Department of Applied Mathematics
University of Johannesburg
South Africa

October 12, 2011

Abstract

We describe an algorithm for controlling the relative error in the numerical evaluation of a bivariate integral, without prior knowledge of the magnitude of the integral. In the event that the magnitude of the integral is less than unity, absolute error control is preferred. The underlying quadrature rule is positive-weight interpolatory and composite. Some numerical examples demonstrate the algorithm.

Keywords: Bivariate integral; Composite quadrature; Interpolatory quadrature; Cubature; Relative error; Absolute error; Error control
MSC 2010: 65D30; 65D32; 65G20

1 Introduction

We consider the evaluation of

$$I\left[G\left(x,y\right)\right]\equiv\int\limits_{a}^{b}\int\limits_{l(x)}^{u(x)}G\left(x,y\right)dydx$$

using cubature based on composite interpolatory quadrature, such that

$$\left|\frac{I\left[G\left(x,y\right)\right]-Q_{C}\left[G\left(x,y\right)\right]}{I\left[G\left(x,y\right)\right]}\right|\leqslant\varepsilon, \tag{1}$$

where $Q_C[G(x,y)]$ is the composite cubature of $G(x,y)$, ε is a user-imposed tolerance, and an estimate of $I[G(x,y)]$ is *not* known a priori. In other words, we seek to control the relative error in the cubature, without prior estimation of the integral. The problem is easily understood with reference to (1): we have

$$|I[G(x,y)] - Q_C[G(x,y)]| \leqslant \varepsilon |I[G(x,y)]|,$$

so that $\varepsilon |I[G(x,y)]|$ is an absolute tolerance. In principle, interpolatory methods readily admit absolute error control but, since $I[G(x,y)]$ is not known, we cannot impose $\varepsilon |I[G(x,y)]|$ as a tolerance. Controlling the relative error is appropriate when dealing with integrals of large magnitude; for such integrals, absolute error control can be very inefficient. Again, however, this presents a problem, since the magnitude of $I[G(x,y)]$ is not known, so we do not even know whether absolute or relative error control should be applied.

The algorithm we present here is, in a sense, a 'first-principles' method, since it is based entirely on classical concepts relating to interpolatory quadrature. The list of references [1-9] is our bibliography, and is drawn from the established literature.

A note regarding terminology: *quadrature* refers to the numerical approximation of a univariate integral, and *cubature* refers to the numerical approximation of a multivariate integral. Both terms will be used throughout this paper.

2 The Algorithm

We transform $[a,b]$ to $[0,1]$ by means of

$$x = (b-a)w + a \equiv m_1 w + a, \tag{2}$$

where $x \in [a,b]$, $w \in [0,1]$ and m_1 has been implicitly defined.

If

$$\begin{aligned} l_1 &\equiv \min_{[a,b]} \{l(x), u(x)\} \\ u_1 &\equiv \max_{[a,b]} \{l(x), u(x)\} \end{aligned}$$

then the transformation between $[l_1, u_1]$ and $[0,1]$ is given by

$$y = (u_1 - l_1) z + l_1 \equiv m_2 z + l_1, \tag{3}$$

where $y \in [l_1, u_1], z \in [0,1]$ and m_2 has been implicitly defined.

As a result of these affine transformations,

$$\int_a^b \int_{l(x)}^{u(x)} G(x,y)\, dydx = \int_0^1 \int_{\widetilde{l}(w)}^{\widetilde{u}(w)} \widetilde{G}(w,z)\, m_1 m_2 dzdw,$$

where

$$\begin{aligned} \widetilde{G}(w,z) &\equiv G(m_1 w + a, m_2 z + l_1) \\ \widetilde{u}(w) &\equiv \frac{u(m_1 w + a) - l_1}{m_2} \\ \widetilde{l}(w) &\equiv \frac{l(m_1 w + a) - l_1}{m_2}. \end{aligned}$$

We determine

$$M \equiv \max\left\{1, \max_{\widetilde{R}} \left|\widetilde{G}(w,z)\, m_1 m_2\right|\right\}, \tag{4}$$

where $\widetilde{R}$ is the domain of integration defined by the transforms (2) and (3). Note that $\widetilde{R} \subseteq [0,1] \times [0,1]$. Hence, we define

$$g(w,z) \equiv \frac{\widetilde{G}(w,z)\, m_1 m_2}{M}.$$

Now,

$$\begin{aligned} & |I[g(w,z)] - Q_C[g(w,z)]| \leqslant \varepsilon \\ \Rightarrow\ & |MI[g(w,z)] - MQ_C[g(w,z)]| \leqslant M\varepsilon \\ \Rightarrow\ & \left|I\left[\widetilde{G}(w,z)\, m_1 m_2\right] - Q_C\left[\widetilde{G}(w,z)\, m_1 m_2\right]\right| \leqslant M\varepsilon \\ \Rightarrow\ & \left|I\left[\widetilde{G}(w,z)\, m_1 m_2\right] - Q_C\left[\widetilde{G}(w,z)\, m_1 m_2\right]\right| \leqslant \left|\frac{I\left[\widetilde{G}(w,z)\, m_1 m_2\right]}{I[g(w,z)]}\right| \varepsilon \\ \Rightarrow\ & \frac{\left|I\left[\widetilde{G}(w,z)\, m_1 m_2\right] - Q_C\left[\widetilde{G}(w,z)\, m_1 m_2\right]\right|}{I\left[\widetilde{G}(w,z)\, m_1 m_2\right]} \leqslant \frac{\varepsilon}{|I[g(w,z)]|} \\ \Rightarrow\ & \frac{|I[G(x,y)] - Q_C[G(x,y)]|}{I[G(x,y)]} \leqslant \frac{\varepsilon}{|I[g(w,z)]|} \approx \frac{\varepsilon}{|Q_C[g(w,z)]|}. \end{aligned}$$

In the last inequality, we use the fact that changes in variable preserve the value of both the integral and the quadrature-based cubature.

Clearly, from the last inequality,

$$\frac{\varepsilon}{\left|Q_C\left[g\left(w,z\right)\right]\right|}$$

is an estimated bound on the relative error in $Q_C\left[G\left(x,y\right)\right] = MQ_C\left[g\left(w,z\right)\right].$ This estimate is good if $Q_C\left[g\left(w,z\right)\right]$ is accurate which, in turn, is determined by the choice of ε.

Now, assuming $\left|I\left[G\left(x,y\right)\right]\right| \geqslant 1$,

$$\frac{1}{\left|Q_C\left[g\left(w,z\right)\right]\right|} = \frac{M}{\left|Q_C\left[G\left(x,y\right)\right]\right|} \approx \frac{M}{\left|I\left[G\left(x,y\right)\right]\right|}$$

and, since M is the maximum possible value of $\left|I\left[G\left(x,y\right)\right]\right|$ (by construction, see (4)), we have

$$\frac{\varepsilon}{\left|Q_C\left[g\left(w,z\right)\right]\right|} \sim \varepsilon,$$

provided $\left|I\left[G\left(x,y\right)\right]\right|$ is not substantially smaller than M. For many practical situations, this will be the case. However, we have no prior knowledge of $\left|Q_C\left[g\left(w,z\right)\right]\right| \approx \left|I\left[g\left(w,z\right)\right]\right|$, so we must be willing to accept the estimate, whatever it may be. Obviously, we cannot expect that the relative error will satisfy the tolerance ε, even if $\left|I\left[g\left(w,z\right)\right] - Q_C\left[g\left(w,z\right)\right]\right|$ does. Note that if

$$\left|I\left[g\left(w,z\right)\right] - Q_C\left[g\left(w,z\right)\right]\right| = \varepsilon,$$

then $\frac{\varepsilon}{\left|Q_C\left[g\left(w,z\right)\right]\right|}$ is not merely an upper bound, but is a very good estimate of the relative error itself.

If $\left|I\left[G\left(x,y\right)\right]\right| < 1$, then the relative error could be considerably larger than ε, particularly if $\left|I\left[G\left(x,y\right)\right]\right| \sim 0$, but in this case we favour absolute error control (for reasons to be discussed later), and so the relative error is not relevant. The quantity $M\varepsilon$ is an upper bound on the absolute error.

If the estimate of the absolute or relative error is considered too large, say by a factor of η, then we simply redo the calculation, this time with a tolerance of

$$\frac{\varepsilon}{\eta}.$$

This refinement is a very important feature of the algorithm, since it enables a desired tolerance to be achieved in a controlled manner, even if it requires a repetition of the calculation. We are sure that such repetition is a small price to pay for a solution of acceptable quality.

3 Bivariate composite interpolatory cubature

Here, we briefly describe bivariate composite interpolatory cubature, including the relevant error analysis. We will consider the effect of roundoff error on error control, and offer a criterion for choosing between absolute and relative error control. A reasonable degree of familiarity with interpolatory quadrature is assumed.

3.1 The form of bivariate composite interpolatory cubature

The composite quadrature that approximates the univariate integral

$$\int_a^b G(x)\,dx$$

is given by

$$Q_C[G(x)] = \sum_{i=1}^{N} c_i G(x_i) = h\sum_{i=1}^{N} w_i G(x_i),$$

where the x_i are nodes on $[a,b]$, the coefficients c_i are appropriate *weights,* h is a stepsize parameter representing the separation of the nodes, and the *reduced* weights are $w_i = c_i/h$.

The bivariate integral

$$\int_a^b \int_{l(x)}^{u(x)} G(x,y)\,dydx$$

is approximated by

$$Q_C[G(x,y)] = h\sum_{i=1}^{N_1} w_i \left(k_i \sum_{j=1}^{N_{2,i}} v_{j,i} G(x_i, y_{j,i}) \right), \tag{5}$$

where $v_{j,i}$ are appropriate reduced weights, $y_{j,i}$ are nodes along the y-axis on $[l(x_i), u(x_i)]$, and k_i are stepsizes, with

$$k_i = \frac{u(x_i) - l(x_i)}{N_{2,i}}.$$

Clearly, bivariate cubature is based on univariate quadrature. We can write

$$Q_C\left[G\left(x,y\right)\right] = \sum_{i=1}^{N_1}\sum_{j=1}^{N_{2,i}} C_{j,i} G\left(x_i, y_{j,i}\right), \tag{6}$$

where

$$C_{j,i} \equiv h w_i k_i v_{j,i}.$$

3.2 Approximation error

The approximation error in $Q_C\left[G\left(x\right)\right]$ is bounded by

$$A\left(r\right)\left(b-a\right)h^r \max_{[a,b]}\left|G^{(r)}\right|,$$

where $A\left(r\right)$ is a numerical constant particular to the type of quadrature used (e.g. Trapezium, Simpson, Gauss-Legendre), and r indicates the so-called *order* of the quadrature. Hence, for bivariate cubature we have

$$A\left(r\right)\left(b-a\right)\underbrace{\max\left(u\left(x\right)-l\left(x\right)\right)}_{D}\left(h^r \max\left|\frac{\partial^r G}{\partial x^r}\right| + \left(\max k_i^r\right)\max\left|\frac{\partial^r G}{\partial y^r}\right|\right)$$

as an upper bound on the approximation error. The integers N_1 and $N_{2,i}$ in (6) can be determined by setting $h = \max k_i$ in the above bound, and demanding

$$h^r A\left(r\right)\left(b-a\right)D\left(\max\left|\frac{\partial^r G}{\partial x^r}\right| + \max\left|\frac{\partial^r G}{\partial y^r}\right|\right) \leqslant \varepsilon$$

$$\Rightarrow \quad h = \left(\frac{\varepsilon}{A\left(r\right)\left(b-a\right)D\left(\max\left|\frac{\partial^r G}{\partial x^r}\right| + \max\left|\frac{\partial^r G}{\partial y^r}\right|\right)}\right)^{\frac{1}{r}}, \tag{7}$$

where the various maxima are found on the region of integration. Then

$$\begin{aligned} N_1 &= \left\lceil \frac{b-a}{h} \right\rceil \\ N_{2,i} &= \left\lceil \frac{u\left(x_i\right)-l\left(x_i\right)}{k} \right\rceil. \end{aligned} \tag{8}$$

Furthermore, the stepsizes h and k must be recalculated to be consistent with integer values of N_1 and $N_{2,i}$, as in

$$\begin{aligned} h^* &= \frac{b-a}{N_1} \\ k_i^* &= \frac{u\left(x_i\right)-l\left(x_i\right)}{N_{2,i}}, \end{aligned} \tag{9}$$

and it is these stepsizes that are used in (5). Once the stepsizes have been determined, the nodes x_i and $y_{j,i}$ can be found.

This process of computing stepsizes consistent with a tolerance ε constitute *absolute* error control in bivariate composite interpolatory cubature, and is used in the previously described algorithm to find $Q_C\left[g\left(w,z\right)\right]$ such that

$$\left|I\left[g\left(w,z\right)\right]-Q_C\left[g\left(w,z\right)\right]\right|\leqslant\varepsilon.$$

It should be noted that our use of $\max\left|\frac{\partial^r G}{\partial x^r}\right|+\max\left|\frac{\partial^r G}{\partial y^r}\right|$ is conservative, and could result in smaller stepsizes than is necessary, for the given tolerance. However, in these types of numerical calculations it is always better to err on the side of caution. Nevertheless, we should be aware that such a conservative approach could result in $\left|I\left[g\left(w,z\right)\right]-Q_C\left[g\left(w,z\right)\right]\right|\ll\varepsilon$, so that $\frac{\varepsilon}{\left|Q_C\left[g\left(w,z\right)\right]\right|}$ overestimates the relative error. Analytically speaking, the approximation error is proportional to

$$\left.\frac{\partial^r G}{\partial x^r}\right|_{(\xi,\zeta)}+\left.\frac{\partial^r G}{\partial y^r}\right|_{(\varphi,\phi)},$$

where (ξ,ζ) and (φ,ϕ) are points somewhere in the region of integration - but since these points are not known, and we cannot be sure of the sign of the derivatives, we use $\max\left|\frac{\partial^r G}{\partial x^r}\right|+\max\left|\frac{\partial^r G}{\partial y^r}\right|$ in the error term, instead.

3.3 Choosing between absolute and relative error control

From (1) we have

$$\left|I\left[G\left(x,y\right)\right]-Q_C\left[G\left(x,y\right)\right]\right|\leqslant\varepsilon\left|I\left[G\left(x,y\right)\right]\right|,$$

so that relative error control is equivalent to absolute error control with an effective tolerance $\varepsilon\left|I\left[G\left(x,y\right)\right]\right|$. Replacing ε in (7) with $\varepsilon\left|I\left[G\left(x,y\right)\right]\right|$ shows that, if $\left|I\left[G\left(x,y\right)\right]\right|>1$, h would be larger than if the tolerance was simply ε, and if $\left|I\left[G\left(x,y\right)\right]\right|<1$, h would be smaller. Consequently, N_1 and $N_{2,i}$ would be smaller or larger, respectively. Smaller values of N_1 and $N_{2,i}$ imply greater computational efficiency and so, for the sake of efficiency, we choose relative error control when $\left|I\left[G\left(x,y\right)\right]\right|>1$, and absolute error control when $\left|I\left[G\left(x,y\right)\right]\right|<1$. When $\left|I\left[G\left(x,y\right)\right]\right|=1$, the two cases are identical. This is why we can impose absolute error control on $\left|I\left[g\left(w,z\right)\right]-Q_C\left[g\left(w,z\right)\right]\right|$ - by our definition of g, $I\left[g\left(w,z\right)\right]$ is guaranteed to have a magnitude less than or equal to one.

3.4 Roundoff error

It is easily shown (see Appendix) that the roundoff error associated with (5) is bounded by

$$4(b-a)D\mu,$$

where μ is a bound on the magnitude of the machine precision of the finite-precision computing device being used, $|G(x,y)| \leqslant 1$ on the region of integration, and the cubature used is based on *positive-weight* quadrature. In such quadrature, all weights are positive; examples of such quadrature include the Trapezium rule, Simpson's rule and all types of Gaussian quadrature. If the region of integration has unit area, as does $I[g(w,z)]$, then the roundoff error simply has the bound 4μ. The roundoff error represents the minimum achievable accuracy in the cubature approximation, and is incorporated into the error control procedure by replacing the numerator of (7) with

$$\varepsilon - 4(b-a)D\mu.$$

Clearly, it makes no sense to impose a tolerance smaller than the roundoff error. A typical desktop PC has $\mu \sim 10^{-16}$.

4 Numerical examples

4.1 Example I: relative error control

We approximate

$$I[G(x,y)] = \int_1^2 \int_{x^2/5}^{x^3/5} e^{4xy}dydx = 1.92660 \times 10^3$$

using Simpson's rule $\left(r=4, A(r)=\frac{16}{180}\right)$. For ease of presentation we show all numerical values truncated to five decimals or fewer, although all our calculations are performed in double precision. The application of the algorithm to this example will be described in detail. With the transformations (using $u_1 = 8/5, l_1 = 1$)

$$\begin{aligned} x &= w+1, y = \frac{7z}{5}+\frac{1}{5} \\ & \qquad \left(\Rightarrow m_1 = 1, m_2 = \frac{7}{5}\right), \end{aligned}$$

the integral becomes

$$
\begin{aligned}
I\left[G\left(w,z\right)\right] &= \int_{0}^{1}\int_{\widetilde{l}(w)}^{\widetilde{u}(w)} \frac{7}{5}e^{4(w+1)\left(\frac{7z}{5}+\frac{1}{5}\right)}dzdw \\
\widetilde{u}\left(w\right) &= \frac{7\left(w+1\right)^{3}}{25}-\frac{7}{5} \\
\widetilde{l}\left(w\right) &= \frac{7\left(w+1\right)^{2}}{25}-\frac{7}{5}.
\end{aligned}
$$

We find

$$
\begin{aligned}
M &= 5.07104\times 10^{5} \\
\max\left|\frac{\partial^4 g}{\partial w^4}\right| &= 1.67772\times 10^{3} \\
\max\left|\frac{\partial^4 g}{\partial z^4}\right| &= 1.57351\times 10^{4} \\
D &= \max\left(\widetilde{u}\left(w\right)-\widetilde{l}\left(w\right)\right)=\frac{18}{26}.
\end{aligned}
$$

The stepsize h is given by

$$
h=\left(\frac{\varepsilon-4\mu}{\left(\frac{16}{180}\right)(1)\left(\frac{18}{26}\right)\left(\max\left|\frac{\partial^4 g}{\partial w^4}\right|+\max\left|\frac{\partial^4 g}{\partial z^4}\right|\right)}\right)^{\frac{1}{4}}=5.52707\times 10^{-4} \qquad (10)
$$

and so, with $\varepsilon=10^{-10}$,

$$
N_1=1810, h^{*}=5.52486\times 10^{-4}
$$

Here, h^* is the length of each simpson subinterval (which contains three nodes), and there are 1810 such subintervals. Hence, there are 3621 nodes w_i on $[0,1]$ with separation $h^*/2$ (this is the reason for the factor $16=2^4$ in (10)).

The stepsizes k_i^* along the z-axis are found from (8) and (9) for each $i=1,2,\ldots,563$, and we find

$$
\max k_i^{*}=5.52706\times 10^{-4}.
$$

This enables the nodes $z_{j,i}$ $(j=1,2,\ldots,N_{2,i})$ to be found, for each i. As with w_i, the spacing between these nodes is $k_i^*/2$. It must be noted that $N_{2,i}$ could

be zero, in which case k_i^* will be NaN (*not-a-number* in IEEE arithmetic). In such cases, it is appropriate to simply set $k_i^* = 0$.

Composite Simpson quadrature of $g(w,z)$ is performed along the z-axis, for each i, yielding the 3621 quantities $Q_C[g(w_i,z)]$, which have the form

$$Q_C[g(w_i,z)] = \frac{k_i^*}{6}\left[\begin{array}{c} g(w_i,z_{1,i}) + 4g(w_i,z_{2,i}) + 2g(w_i,z_{2,i}) + 4g(w_i,z_{4,i}) + \\ \ldots + 2g\left(w_i,z_{N_{2,i}-2,i}\right) + 4g\left(w_i,z_{N_{2,i}-1,i}\right) + g\left(w_i,z_{N_{2,i},i}\right) \end{array}\right].$$

The integer coefficients in this expression are the weights appropriate to composite Simspon quadrature.

Finally, Simpson quadrature is performed over these quantities along the w-axis, to give

$$\begin{aligned} Q_C[g(w,z)] &= \frac{h_i^*}{6}\left[\begin{array}{c} Q_C[g(w_1,z)] + 4Q_C[g(w_2,z)] + 2Q_C[g(w_3,z)] + \\ 4Q_C[g(w_4,z)] + \ldots + 2Q_C[g(w_{N_1-2},z)] + \\ 4Q_C[g(w_{N_1-1},z)] + Q_C[g(w_{N_1},z)] \end{array}\right] \\ &= 3.79922 \times 10^{-3}. \end{aligned}$$

Hence,

$$I[G(x,y)] \approx MQ_C[g(w,z)] = 1.92660 \times 10^3.$$

The estimate of the relative error is

$$\left|\frac{\varepsilon}{Q_C[g(w,z)]}\right| = 2.63211 \times 10^{-8}$$

while the actual relative error is 1.47276×10^{-11}. Clearly, the actual error is less than the estimate. This is to be expected when using $\max\left|\frac{\partial^r G}{\partial x^r}\right| + \max\left|\frac{\partial^r G}{\partial y^r}\right|$ in the computation of h. Obviously, our value for h is conservative (smaller than actually necessary) and so the actual error is smaller than the estimate. Nevertheless, as we have stated earlier, it is better to be more accurate than necessary and, since the estimate reflects an upper bound, we can be sure that the error is no more than 2.63211×10^{-8}. If this level of accuracy is acceptable, then the result stands. However, if we desire a relative error of no more than 10^{-10}, say, we simply repeat the algorithm with

$$\frac{\varepsilon}{264}$$

as the new tolerance. This gives

$$\left|\frac{\varepsilon}{Q_C[g(w,z)]}\right| = 9.97014 \times 10^{-11} < 10^{-10},$$

while the actual relative error is 5.35801×10^{-14}.

4.2 Example II: absolute error control

In this second example, the integral

$$I\left[G\left(x,y\right)\right]=\int\limits_{1}^{4}\int\limits_{x}^{2x^{2}}\frac{\sin\left(xy\right)}{5}dydx=-0.00734$$

will again be approximated using Simpson quadrature but, since it has magnitude less than one, we will see that absolute error control is more efficient than relative error control. There is no need for a detailed exposition, as in the previous example, and we simply state our results.

The transformed integral is

$$\begin{aligned}
I\left[G\left(w,z\right)\right] &= \int\limits_{0}^{1}\int\limits_{\widetilde{l}(w)}^{\widetilde{u}(w)}\frac{93}{5}\sin\left(\left(3w+1\right)\left(31z+1\right)\right)dzdw \\
\widetilde{u}\left(w\right) &= \frac{2\left(3w+1\right)^{2}-1}{31} \\
\widetilde{l}\left(w\right) &= \frac{3w}{31}.
\end{aligned}$$

Using $M=\frac{93}{5}$ and $\varepsilon=10^{-4}$ gives

$$\begin{aligned}
M\varepsilon &= 0.00186 \\
\left|\frac{\varepsilon}{Q_{C}\left[g\left(w,z\right)\right]}\right| &= 0.25340.
\end{aligned}$$

The upper bound on the relative error is fairly large. The absolute error is estimated by $M\varepsilon$; it is clear that, since M is known, ε can be chosen so that $M\varepsilon$ equals some desired value. For example, if we seek an absolute error of no more than 10^{-5}, we choose $\varepsilon=5.37\times10^{-7}$, which gives

$$\begin{aligned}
M\varepsilon &= 9.9882\times10^{-6}<10^{-5} \\
\left|\frac{\varepsilon}{Q_{C}\left[g\left(w,z\right)\right]}\right| &= 0.00136,
\end{aligned}$$

with $N_1=1761$ (hence, 3523 nodes on the w-axis). Note that achieving this tolerance does not require a repetition of the algorithm, since M is known a priori.

On the other hand, to improve the estimate of the relative error to 10^{-5} requires $\varepsilon=10^{-5}/136=7.353\times10^{-8}$, which results in $N_1=2895$, and hence, more nodes than are needed to achieve the same tolerance in the absolute error. This is consistent with our earlier discussion regarding the efficiency-based criterion for choosing between absolute and relative error control.

5 Conclusion

We have reported on an algorithm for controlling the relative error in the numerical approximation of a bivariate integral. The numerical method used is positive-coefficient composite interpolatory quadrature. The algorithm involves transforming and scaling the integral to one that has magnitude bounded by unity, and then applying an absolute error control procedure to such integral. The relevant scaling factor is then used to find the approximate value of the original integral and an estimate of the relative error (if the integral has magnitude greater than unity) or absolute error (if the integral has magnitude less than or equal to unity). The calculation can be repeated with an appropriate refinement if the estimated error is considered too large. The algorithm proceeds in a systematic and controlled manner, and there is no need for any prior knowledge of the magnitude of the integral. Two examples with Simpson's rule clearly demonstarte the character of the algorithm. This work extends other work of ours [10], in which we considered the control of relative error in the quadrature of a univariate integral. In that work, we designated the algorithm CIRQUE, and so we take the liberty here of designating the current algorithm CIRQUE2D.

References

[1] Burden, R.L., and Faires, J.D. (2011). *Numerical Analysis 9th ed.*, Brooks/Cole.

[2] Davis, P.J., and Rabinowitz, P. (1984). *Methods of Numerical Integration 2nd ed.*, New York: Academic Press.

[3] Engels, H. (1980). *Numerical Quadrature and Cubature*, New York: Academic Press.

[4] Ghizetti, A., and Ossiccini, A. (1970). *Quadrature Formulae*, New York: Academic Press.

[5] Isaacson, E., and Keller, H.B. (1994). *Analysis of Numerical Methods*, New York: Dover.

[6] Kincaid, D., and Cheney, W. (2002). *Numerical Analysis: Mathematics of Scientific Computing 3rd ed.*, Pacific Grove: Brooks/Cole.

[7] Krommer, A.R., and Ueberhuber, C. W. (1998). *Computational Integration*, Philadelphia: SIAM.

[8] Stroud, A.H., (1974). *Numerical Quadrature and Solution of Ordinary Differential Equations*, Berlin: Springer-Verlag.

[9] Stroud, A.H., and Secrest, D. (1966). *Gaussian Quadrature Formulas*, Englewood Cliffs: Prentice-Hall.

[10] Prentice, J.S.C., Absolute and Relative Error Control in Composite Interpolatory Quadrature: the CIRQUE Algorithm, *Journal of Mathematics Research*, 3, 3 (2011) 63-72 [http://www.ccsenet.org/journal/index.php/jmr/article/view/9445/8108]

A Roundoff bound

Using (5) an (6), we write

$$\begin{aligned}
Q_C\left[G\left(x,y\right)\right] &= \sum_{i=1}^{N_1} c_i\left(1+\mu_{c,i}\right) \\
&\quad \times \sum_{j=1}^{N_{2,i}} k_i\left(1+\mu_{k,i}\right) v_{j,i}\left(1+\mu_{v,j,i}\right) G\left(x_i,y_{j,i}\right)\left(1+\mu_{G,j,i}\right) \\
&= \sum_{i=1}^{N_1}\sum_{j=1}^{N_{2,i}} C_{j,i}G\left(x_i,y_{j,i}\right) + C_{j,i}G\left(x_i,y_{j,i}\right)\left(\mu_{w,i}+\mu_{v,j,i}+\mu_{G,j,i}\right),
\end{aligned}$$

where we have indicated the roundoff error in $c_i, k_i, v_{j,i}$ and $G\left(x_i,y_{j,i}\right)$ explicitly, and we have ignored higher-order terms in the second line. The roundoff error Υ in the cubature is

$$\begin{aligned}
\Upsilon &\equiv \sum_{i=1}^{N_1}\sum_{j=1}^{N_{2,i}} C_{j,i}G\left(x_i,y_{j,i}\right)\left(\mu_{c,i}+\mu_{w,i}+\mu_{v,j,i}+\mu_{G,j,i}\right) \\
&\leqslant \sum_{i=1}^{N_1}\sum_{j=1}^{N_{2,i}} 4C_{j,i}\mu,
\end{aligned}$$

where μ is a bound on $\left|\mu_{c,i}\right|, \left|\mu_{w,i}\right|, \left|\mu_{v,j,i}\right|$ and $\left|\mu_{G,j,i}\right|$, and we have assumed $\left|G\left(x_i,y_{j,i}\right)\right| \leqslant 1$. Now, since $C_{j,i} = hw_ik_iv_{j,i} = c_ik_iv_{j,i}$,

$$\Upsilon \leqslant 4\mu\sum_{i=1}^{N_1} c_i\left(\sum_{j=1}^{N_{2,i}} k_iv_{j,i}\right).$$

But, in positive-weight univariate composite interpolatory quadrature, the sum of the weights is simply the length of the interval of integration, and so

$$\begin{aligned} \Upsilon &\leqslant 4\mu\,(b-a)\,(\max\,(u\,(x_i)-l\,(x_i))) \\ &= 4\mu\,(b-a)\,D \\ &\leqslant 4\mu \end{aligned}$$

if $(b-a)\,D \leqslant 1$.